Genetic Algorithms for Tracking Changing Environments

Helen G. Cobb
Navy Center for Applied Research in AI
Naval Research Laboratory, Code 5514
Washington, DC 20375-5337
Internet: cobb@aic.nrl.navy.mil

John J. Grefenstette
Navy Center for Applied Research in AI
Naval Research Laboratory, Code 5514
Washington, DC 20375-5337
Internet: gref@aic.nrl.navy.mil

Abstract

In this paper, we explore the use of alternative mutation strategies as a means of increasing diversity so that the GA can track the optimum of a changing environment. This paper contrasts three different strategies: the Standard GA using a constant level of mutation, a mechanism called Random Immigrants, that replaces part of the population each generation with randomly generated values, and an adaptive mechanism called Triggered Hypermutation, that increases the mutation rate whenever there is a degradation in the performance of the time-averaged best performance. The study examines each of these strategies in the context of several kinds of environmental change, including linear translation of the optimum, random movement of the optimum, and oscillation between two significantly different landscapes. These first results should lead to the development of a single mechanism that can work well in both stationary and nonstationary environments.

1 INTRODUCTION

In nature, diversity helps to ensure a population's survival under changing environmental conditions. In a genetic algorithm (GA), which is a codification of Holland's adaptive model (Holland, 1975), diversity in the population should also be useful in tracking a changing environment, since the population members represents potential solutions that can be applied to different environmental circumstances. However, since Holland's seminal work, GAs have been successfully fine-tuned to perform function optimization on stationary functions (De Jong, 1992). We call this optimizing form of the GA the Standard GA. The Standard GA, which uses a strong selection policy based on scaled environmental feedback and a small mutation rate, quickly eliminates diversity from the population as it seeks out a global optimum. In typical applications, the function representing the environment remains static so that the algorithm's "adaptiveness" is limited to finding a single solution. Should the environment change, the Standard GA is often unable to redirect its search to a different part of the space. Thus, the Standard GA, with its typical parameter settings, has difficulty in tracking a moving optimum over time. An early study by Pettit and Swigger (1983), in which the Standard GA searches for a target string that randomly changes every generation with some small probability, lends some support to this view.

The problem of optimization in a nonstationary environment can be thought of as optimizing a series of time-dependent optima. Because the Standard GA works quickly to find an optimum, some modified version of the Standard GA might be useful in searching for a series of optima. There are two basic strategies for modifying the GA to accommodate changing environments. The first strategy is to expand the memory of the GA in order to build up a repertoire of ready responses for environmental conditions. The second strategy is to employ some method for increasing diversity in the population (e.g., by using mutation) in order to compensate for changes encountered in the environment. In this paper, we focus on the second strategy. We continue an investigation which explores the effectiveness of various mutation schemes in enhancing the Standard GA's ability to track the optimum of a changing environment (Cobb, 1990; Grefenstette, 1992).

There are several previous studies that address the problem of using genetic algorithms in changing environments. The study of Goldberg and Smith (1987) explores the first strategy of expanding each population member's structure. Their experiments show the effectiveness of the Hollstien-Holland triallelic representation, which includes a diploid chromosome and a third allelic structure for deciding dominance, in the context of an environmental optimum that oscillates periodically between two different states. In a more recent study, Dasgupta and McGregor (1992) explore a modified GA, called the Structured Genetic Algorithm (sGA), that uses tree structured population

members. Higher level nodes in the gene structures regulate the activation or de-activation of lower level genes. The sGA successfully tracks the environments of the two state problem explored by Goldberg and Smith. It is not clear how either of these memory modifications scale up when there are a larger number of environmental states.

The micro-GA (μGA) developed by Krishnakumar (1989) uses the second strategy of increasing population diversity. The μGA uses a very small population (5 members). A small population permits the μGA to converge quickly to a local optimum (in both space and time); but this convergence also quickly reduces any diversity in the small population so that partial or complete replacement of the members is necessary almost every generation. The μGA does not take advantage of the diversity of solutions within a larger population.

In a recent study, Grefenstette uses a similar replacement strategy, called *Random Immigrants*, within the context of a larger population (Grefenstette, 1992). The Random Immigrants mechanism replaces a fraction of a Standard GA's population each generation, as determined by the *replacement rate*, with randomly generated values. This mechanism views the GA's population as always having a small flux of immigrants that wander in and out of the population from one generation to the next. Grefenstette's study shows that the Random Immigrants mechanism works well in environments where there are occasional, large changes in the location of the optimum. In another study, Cobb investigates an adaptive mutation-based mechanism, called *Triggered Hypermutation*, which temporarily increases the mutation rate to a high value (called the *hypermutation rate*) whenever the time-averaged best performance of the population deteriorates (Cobb, 1990). Cobb's study shows the ability of the hypermutation mechanism to adjust to changes in continuous, time-dependent nonstationary environments. In this paper, we continue to investigate the Random Immigrants and Triggered Hypermutation mechanisms. (The respective GAs are called the Random Immigrants GA and the Hypermutation GA for convenience.) In addition, we explore the effect of simply increasing the mutation rate within the Standard GA to a constant high level. This study represents the first systematic exploration comparing these mechanisms.

The remainder of this paper is organized as follows: In Section 2, we present the GA modifications, the types of changing environments explored, and the methodology used in making comparisons. Section 3 presents a series of graphs which illustrate the results; Section 4 gives conclusions and a brief description of future research plans.

2 EXPERIMENTAL DESIGN

The study compares three modifications to the Standard GA on tracking in changing environments. The modifications include:

(1) the Random Immigrants mechanism,

(2) increasing the mutation rate in the Standard GA, and

(3) the Triggered Hypermutation mechanism.

In the Random Immigrants GA, the *replacement rate* specifies the fraction of the population that is replaced each generation by randomly generated strings. This strategy effectively concentrates mutation in a subpopulation while maintaining a traditionally low (i.e., 0.001) mutation rate in the remainder of the population.

The use of an overall mutation rate in the Standard GA examines the effect of distributing mutation uniformly throughout the population.

The Triggered Hypermutation mechanism also uses uniformly distributed mutation throughout the population at any point in time; however, the distribution of mutation levels is not uniform from a temporal point of view. When the adaptive mechanism is triggered due to a degradation of performance, the level of mutation is high (the *hypermutation rate*); otherwise, the Hypermutation GA uses a low baseline mutation rate of 0.001. In this study, each generation's performance is measured as the running average of the best performing population members over a period of five generations. Table 1 summarizes the types of GAs considered here.

Obviously, there is an infinite variety of changing environments, and it is not possible to do an exhaustive study of all cases. For this study, we limit our attention to a suite of test problems based on two underlying surfaces: Landscape A, shown in Figure 1, consists of 14 sinusoidally shaped hills.

Table 1: Mutation Mechanisms in GAs

Type of GA	**Description**	**Mutation Parameter**
Standard	Uniformly distributed probability of mutation every generation	Mutation rate
Random Immigrants	Non-uniformly distributed probability of mutation every generation	Replacement rate
Hyper-mutation	Temporally non-uniform distribution of probability of mutation; uniform within each generation	Hyper-mutation rate

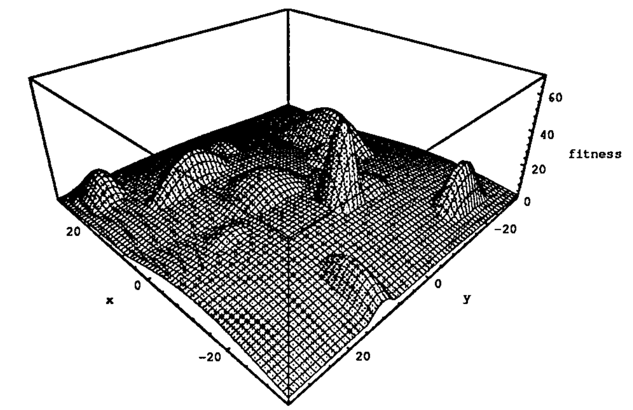


Figure 1: Landscape A.

Generated Using 14 Sinusoidally Shaped Hills

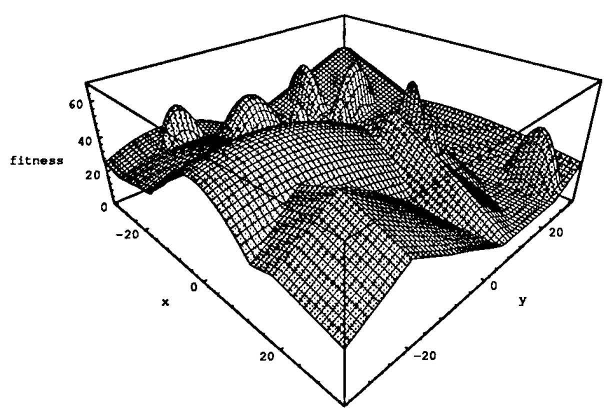


Figure 2: Landscape B.

A Combination of 20 Gaussian, Triangular, and Sinusoidally Shaped Hills

Landscape B, shown in Figure 2, consists of 20 hills that are shaped using a sine function, and triangular and Gaussian probability distributions. For convenience in measuring the performance of the algorithms, the maximum in both of these landscapes is always a height of 60. These functions are two-dimensional, each dimension ranging from approximately -32.768 to 32.768, with 16 bits used to specify values within each dimension. In other words, each population member is 32 bits long.

We consider three main categories of environmental change, including:

(1) linear translation of all of the hills in Landscape A;

(2) randomly mutating the location of the maximum hill in Landscape A every 20 generations while keeping the remainder of the landscape fixed; and

(3) oscillating the environment between Landscape A and Landscape B so that the entire landscape changes significantly.

In addition to these changing environments, we also examine the three mutation mechanisms in a stationary environment using Landscape A.

We also consider several subcases of the above categories. For the first category, there are two subcases: slow translation and fast translation. In the slow mode, all hills in Landscape A translate at the rate of +0.2 or -0.2. In the fast mode, all hills translate at the rate of +0.5 or -0.5. For example, a rate of +0.2 means that the hill's location increases by one step in both dimensions after 5 generations. Each dimension's rate of change is specified independently, so that one dimension can increase while another decreases.

There are also two subcases for the third category of environmental change: a fast mode in which an oscillation occurs every 2 generations, and a slow mode in which an oscillation occurs every 20 generations.

Our investigations were performed in two stages.[1] In the first stage, we made a set of preliminary runs covering a spectrum of different mutation levels for each combination of mutation mechanism and category of environmental change. Each combination used the same set of seeds for a given repetition and generation, in order to factor out the effects of variables other than mutation level, mutation mechanism, and environmental change. For the Random Immigrants mechanism, replacement rate settings of 0.1, 0.2, 0.5, 0.55, 0.6, 0.65, ..., 0.95, and 0.99 were examined for each of the categories of environmental change. The same levels were used to specify the mutation rate within the Standard GA and the hypermutation rates in the Hypermutation GA.

In the second stage, we repeated selected combinations 35 times in order to make statistical comparisons on a generation-by-generation basis.[2] Based on the results of the first stage runs, the repetitions of combinations use the levels 0.01, 0.02, 0.05, 0.10, 0.15, 0.20, 0.25, 0.30, and 0.40.These levels were found to be representative of the

[1] All runs use variations of GENESIS, Version 5.0, written by J. Grefenstette (Grefenstette, 1984).

[2] A non-parametric randomization test for dependent samples is used so that no distribution assumptions need to be made (Krauth, 1988).

best overall performance of the various GAs. In all cases, the population size was 100, the crossover rate was 0.6, the crossover operator was two-point crossover, and the convergence testing mechanisms within the GA were disabled.

In the next section, we summarize some of the results of the second phase. In all graphs, we use the best parameter settings for the given GA for the particular class of environmental change being considered. This approach is intended to minimize the possibility that one form of GA might appear to outperform another based on unfavorable parameter settings.

3 COMPARATIVE STUDIES

3.1 STATIONARY ENVIRONMENT

We begin by comparing the three GAs on a stationary function. Figure 3 shows the characteristic behavior of the three mutation strategies for a stationary environment. The levels selected indicate the best overall performance for each of the strategies: a hypermutation rate of 0.01 for Triggered Hypermutation; a mutation rate of 0.02 for the Standard GA, and a replacement rate of 0.05 for Random Immigrants. The line marked with squares shows the best performance each generation; the line marked with plus signs shows the average population performance.

The Triggered Hypermutation mechanism shows little degradation in the average performance each generation because the system does not enter hypermutation very often; the mutation level tends to remain at the baseline rate. Occasionally, however, the mechanism erroneously responds to a perceived degradation when there is not any. This result emphasizes the fact that Triggered Hypermutation is an adaptive strategy, whereas a constant high mutation level in the Standard GA and the use of the Random Immigrants mechanism always contribute randomness to the population. Nevertheless, in all three cases the best performance is high.

3.2 TRANSLATING VERSUS ARBITRARY PERIODIC RELOCATION OF THE MAXIMUM HILL

Next we consider the performance of GAs on changing environments. Both graphs of Figure 4 show the median performance out of 35 repetitions for the average and best performance of the GAs. For the first 50 generations, hills are linearly translated at a slow rate; for the subsequent 50 generations, hills are translated at a fast rate, and finally, for the last 100 generations, the location of the maximum is randomly changed on a periodic basis every 20 generations. The top figure compares the performances of the Standard GA and the Random Immigrants GA.

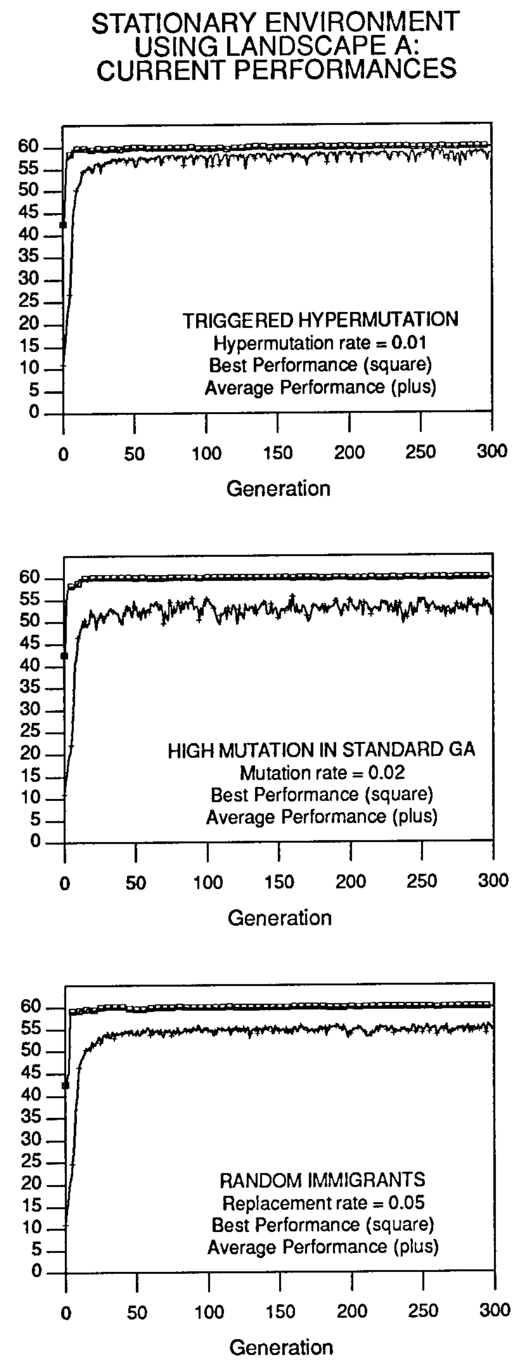
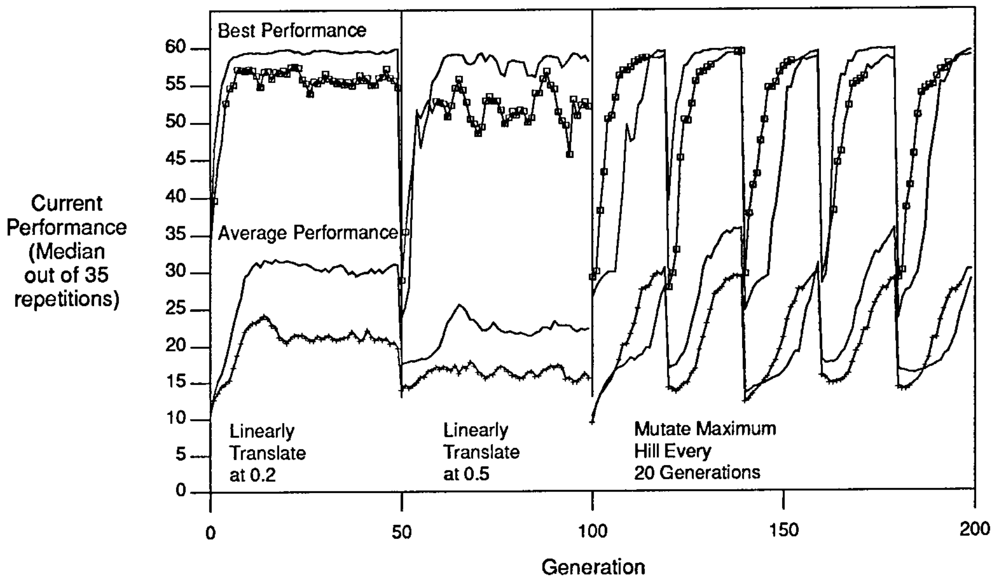


Figure 3: Comparison of Best and Current Average Performance for the Three Mechanisms

LANDSCAPE A: FOURTEEN SINUSOIDALLY SHAPED HILLS

STANDARD GENETIC ALGORITHM WITH MUTATION RATE = 0.10 AND
RANDOM IMMIGRANTS AT REPLACEMENT RATE = 0.40
(□ and + indicate generations on the Random Immigrants line
that are significantly different from the Standard GA)



STANDARD GENETIC ALGORITHM WITH MUTATION RATE = 0.10 AND
HYPERMUTATION GENETIC ALGORITHM
(□ and + indicate generations on the Hypermutation line
that are significantly different from the Standard GA)

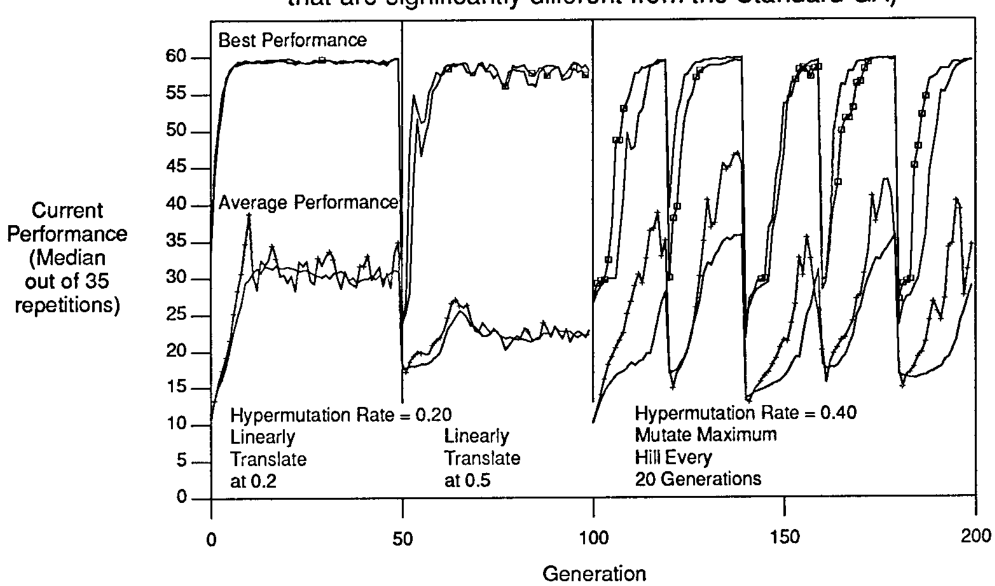


Figure 4: Combination of Translation followed by Mutation of the Maximum Hill's Location

The Random Immigrants GA performs its best for all three cases of environmental change when the replacement rate is 0.40. For the Standard GA, a mutation rate of 0.10 provides its overall best performance. The square marks indicate generations where the best performance of the Random Immigrants GA differs statistically from the Standard GA's best performance. Similarly, the plus signs indicate generations where the average performance of the Random Immigrants GA is significantly different than Standard GA's.

The bottom figure shows a similar comparison of the Standard GA with the Hypermutation GA (marked by squares and pluses). Again, the Standard GA's mutation rate is held constant at 0.10. The hypermutation rate needs to be at least 0.20 to provide competitive performance during translation. When tracking more abrupt changes in the environment, a greater hypermutation rate of 0.40 is required.

Both the Hypermutation GA and the Standard GA perform better than the Random Immigrants GA for the translation cases. Regardless of the rates selected for the Random Immigrants GA (out of 0.5 through 0.40 by increments of 0.05) or the Standard GA, the Standard GA tracks the optimum better (i.e., the Standard GA has the higher best performance). The average performance of the Standard GA drops substantially when the mutation level is increased as one would expect. The Hypermutation GA only performs as well as the Standard GA if the hypermutation rate is sufficiently high. For lower hypermutation rates, the performance is generally worse than that of the Random Immigrants GA.

For the case where the location of the maximum hill changes every 20 generations, the Standard GA performs as well as the Random Immigrants GA when the mutation is 0.10. (Notice that the squares indicating any statistical difference sometimes fall above and sometimes fall below the Standard GA's line.) The Standard GA and the Hypermutation GA also perform comparably, provided the hypermutation rate is sufficiently high. However, the Hypermutation GA has more variance in its tracking than either the Standard GA or the Random Immigrants GA.

3.3 OSCILLATING ENVIRONMENTS

Figure 5 shows the best performance of the three GAs on for a case where the environment oscillates between Landscape A and Landscape B. For the first 100 generations, the oscillation between the two landscapes occurs every 2 generations; for the last 100 generations, the oscillation occurs every 20 generations. A mutation rate of 0.15 provides the best performance for the Standard GA for the fast mode oscillation; a mutation of 0.05 does best for the slow mode. A replacement rate of 0.10 provides the best performance for the Random Immigrants GA when the oscillation is rapid; a replacement rate of 0.05 is best when the oscillation is slower.

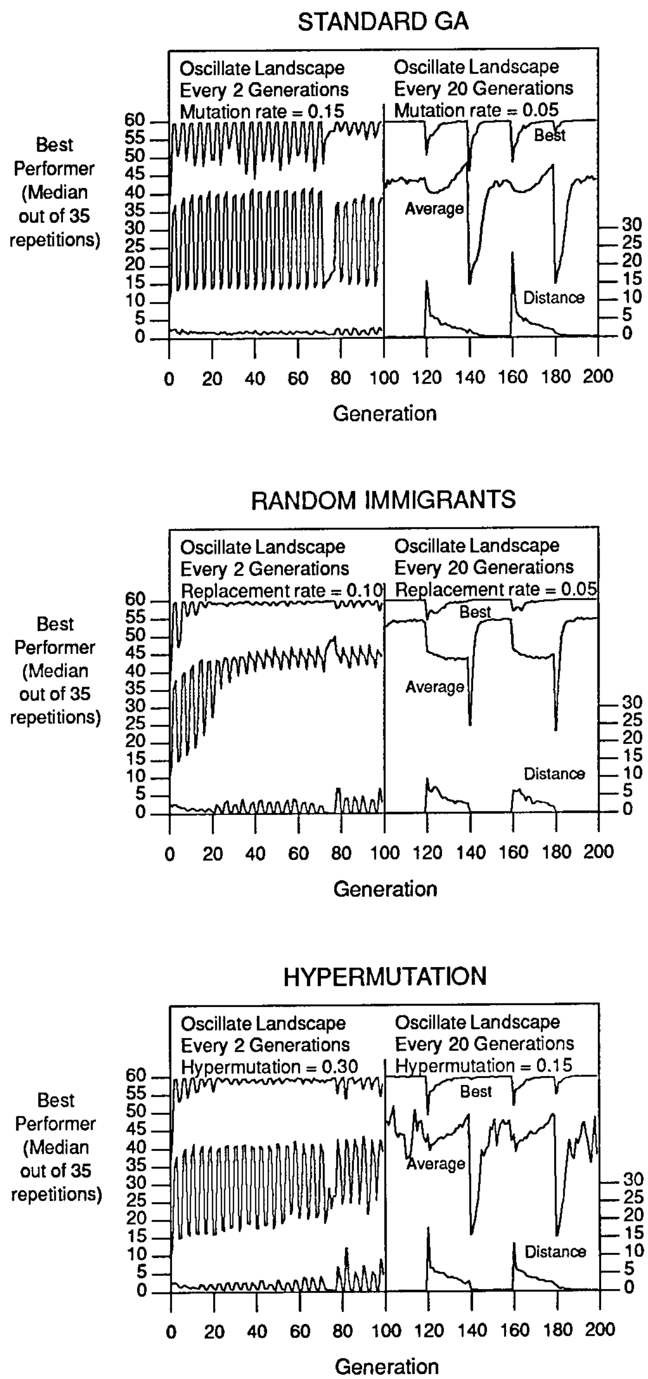


Figure 5: Oscillation Between Two Different Landscapes

Note that Hypermutation requires a much higher rate to track this environment -- 0.30 for the rapid oscillation and 0.15 for the slower case. Overall, the Random Immigrants GA tracks the optimum best, especially when considering average performances. The oscillation in the tracking performance is smaller and the recovery time is fastest for the Random Immigrants GA. In addition, the replacement rate required is generally quite small. The small replacement rate allows the Random Immigrants GA to add enough diversity to the population so that it can adjust to changes in the environment, while at the same time preserving enough of the population so that the GA can find the optimum of a more complicated landscape (Landscape B). These results are generally consistent with those reported in (Grefenstette, 1992).

4 CONCLUSIONS

This paper presents an initial attempt at a systematic comparison of three different mutation strategies for enabling the Standard GA to track a changing environment's optima. These studies show that diversity represents a natural source of power in adapting to changing environments. There are advantages and disadvantages to each of the approaches, depending on the type of environmental change.

The Standard GA at high mutation levels (i.e., 0.10) provides good tracking performance for environments that change continuously through translation, but with this overall increase in mutation, the average (online) performance deteriorates. It seems important to match the level of mutation with the degree of change going on in the environment. This may limit the usefulness of this approach when the degree of environmental change is unknown.

Triggered Hypermutation has the advantage of being adaptive; for certain classes of environmental change, the Hypermutation GA adaptively introduces diversity when needed. However, as reported in (Grefenstette, 1992), the mechanism sometimes does not perform well in abruptly changing environments. In other cases, the level of mutation may exceed the amount required. As a result, the Hypermutation GA exhibits more variance in tracking continuously changing environments.

The Random Immigrants GA introduces randomness into a percentage of its population, and seems to prepare the GA well for a possible catastrophic change in the environment. However, this mechanism incurs a constant cost in a stationary environment. In addition, this approach increases the probability of losing information that may match small incremental changes in the environment, as shown by the relatively poor performance on the translating environments case.

For oscillating environments where the changes are significant, the Random Immigrants mechanism is more conservative in its use of mutation than Triggered Hypermutation. This mutation mechanism permits some preservation of information in part of its population so that when the environment returns to a prior state it can find the optimum fairly quickly. At the same time, the random immigrants entering the population permit additional diversity so that the GA can potentially track more than one optimum.

In addition to the landscapes described here, we have designed an extensive suite of other test cases. Future studies will attempt to verify the initial observations reported here on a larger variety of changing environments.

In practice, one may not know whether or how an environment changes over time. Our eventual goal is to find a GA that works well on a variety of nonstationary environments as well as with stationary ones. Hybrid mechanisms will be examined in future studies. For example, adding an adaptive mechanism to the Random Immigrants GA and expanding its memory in some way might be fruitful. One method for expanding the GA's memory would be to save the subpopulation removed by the random immigrants so that they could be reintroduced during future generations. Also, it might be beneficial to apply different mutation levels to more than one subpopulation within the GA. Finally, the Hypermutation mechanism might be improved by using a different kind of hypermutation "trigger." We will report on the effectiveness of these approaches in future articles.

References

H. G. Cobb (1990). An investigation into the use of hypermutation as an adaptive operator in genetic algorithms having continuous, time-dependent nonstationary environments. *NRL Memorandum Report 6760.*

Dipankar Dasgupta and Douglas R. McGregor (1992). Nonstationary Function Optimization using the Structured Genetic Algorithm. *Parallel Problem Solving from Nature,* 2, pp. 145-154. (Reinhold Manner and Bernard Manderick, eds.). Amsterdam: North Holland.

Kenneth A. De Jong (1975). *An Analysis of the Behavior of a Class of Genetic Adaptive Systems.* Doctoral Dissertation, Univ. of Michigan.

K. A. De Jong (1992). Are Genetic Algorithms Function Optimizers? *Parallel Problem Solving from Nature,* 2. pp. 3 - 14. (R. Manner and B. Manderick, eds.). Amsterdam: North Holland.

David E. Goldberg and Robert E. Smith (1987). Nonstationary function optimization using genetic dominance

and diploidy. *Proceedings of the Second International Conference on Genetic Algorithms*, pp. 59 - 68. Hillsdale, New Jersey: Lawrence Erlbaum Associates.

John J. Grefenstette (1984). GENESIS: a system for using genetic search procedures. *Proceedings of a Conference on Intelligent Systems and Machines*, pp. 161-165. Rochester, MI.

John J. Grefenstette (1992). Genetic algorithms for changing environments. *Parallel Problem Solving from Nature, 2*. pp. 137 - 144. (Reinhard Manner and Bernard Manderick, eds.). Amsterdam: North Holland.

John Holland (1975). *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI: The University of Michigan Press.

Joachim Krauth (1988). *Distribution-Free Statistics: An Application-Oriented Approach*. The Netherlands: Elsevier Science Publishers B. V.

K. Krishnakumar (1989). Micro-genetic algorithms for stationary and non-stationary function optimization. *SPIE, Intelligent Control and Adaptive Systems*, pp. 289-296.

K. Pettit and E. Swigger (1983). An analysis of genetic based pattern tracking and cognitive-based component tracking models of adaptation. *Proceedings of National Conference on AI (AAAI-83)*, pp. 327-332. Morgan Kaufmann.